

Automata Architects: Essay

Josh Humphriss

January 2026

1 Introduction

A Discrete Finite Automata (DFA) [1] is a basic model of computation that computes regular languages. DFAs form an integral part of defining classes of languages and the notion of what is computable, and is therefore a part of nearly every computer science degree course globally. In particular, students need to have a good understanding of what a DFA is, how it works, and the ability to quickly construct DFAs to recognise any language. *Automata Architects* is a tabletop game for 2 players aimed to help practise these skills either as an in-class activity or to reinforce the learning outside of the lesson. More details about what a DFA is and the learning objectives can be found in the rulebook.

Definition (Word). *A word is a sequence of bits (0 or 1). Example: 001.*

Definition (Language). *A language is a set of words. Example: all words starting with 0.*

2 Learning Objectives

The learning objectives are aligned with University modules such as CS143 at Warwick [3]. Bloom's Taxonomy [6] (table 1) shows that the learning objectives are varied and stretching.

Learning outcomes

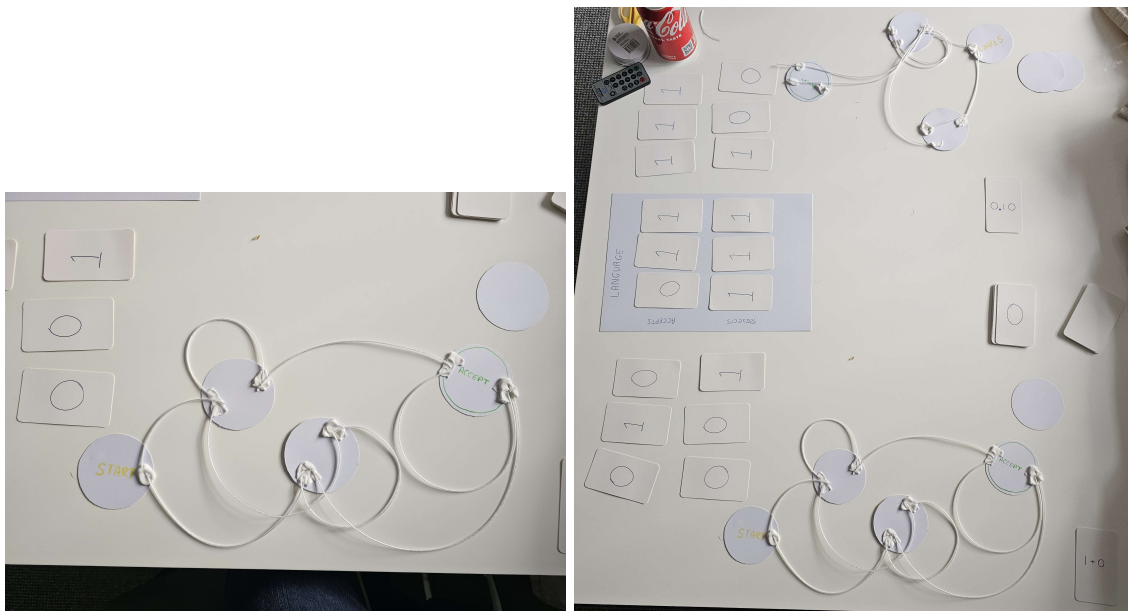
By the end of the module, students should be able to:

- Construct and reason about proofs in a variety of logics.
- Apply logic to specify and verify computing systems.
- Specify formal languages and translate between various forms of formal language descriptions.
- Argue that given formal languages are (or are not) regular.
- Understand basic connections between expressive power of some logics and computational models.

Figure 1: CS143 [3] Learning Objectives. This game satisfies part of points 3 and 4.

Learning Objective	Bloom's Taxonomy
Understand what a DFA is and how it works.	Understanding
Quickly simulate a DFA to determine if a...	Applying
Adapt existing DFAs to recognise new...	Applying/Creating
Construct DFAs from regexes.	Creating

Table 1: Bloom's taxonomy for the learning objectives.



(a) A single DFA.

(b) The whole game.

Figure 2: Prototype 1 in action.

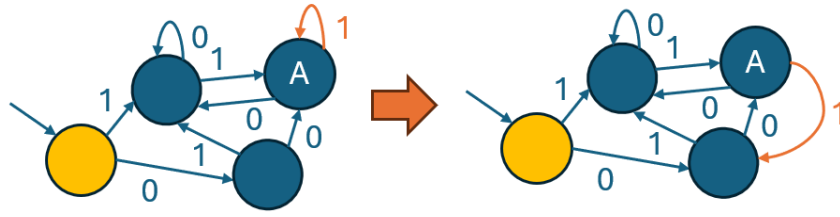


Figure 3: An example movement within a DFA.

3 Prototype 1

Figure 2 depicts the first prototype.

3.1 DFA Manipulation

While it is possible for players to create a new DFA each time, this is repetitive and unengaging, offering little more value than a worksheet. Therefore, players will make modifications to an existing DFA. This requires players to plan ahead and engage with the concepts on a deeper level. DFAs are modified by moving one of the outputs to a different state, as this always leads to a valid DFA after every step (see figure 3 for an example). One move per turn was permitted.

3.2 Languages

Instead of requiring a whole language, language cards will instead provide one or two languages to accept/reject, so that multiple can be recognised at once and the information is not overwhelming.

Players have 3 languages to recognise, formed of tiles laid out on a mat (envisaged to be like Mahjong [7] tiles). When recognising a language, it was removed and replaced with a new one. If the same word was in both accepting and rejecting, both words were re-done. Some mats would have more slots for binary tiles to give more complex languages.

While this led to maximum possibilities, it was hugely impractical and dominated the game time.

3.3 Go-Again Mechanic

To introduce extra excitement, if a player recognised a language they would immediately receive another turn. This was designed to make recognising languages more fun and rewarding. However, it turned out disastrous, as the majority of the game time was spent with only one player playing. Sometimes, it would be possible to recognise 10 languages in a row through sheer chance only for the opponent to then do one move and pass play back. It is clear the game is more fun if players are taking equal turns and recognising more languages.

3.4 Starting Minigame

Many games have a fun way to decide who goes first. However, we also want to force the DFAs into an initial state. If players could choose their starting state, experienced players will memorise particular strategies, as with chess openings [9], which allows the game to get stale. Constructing

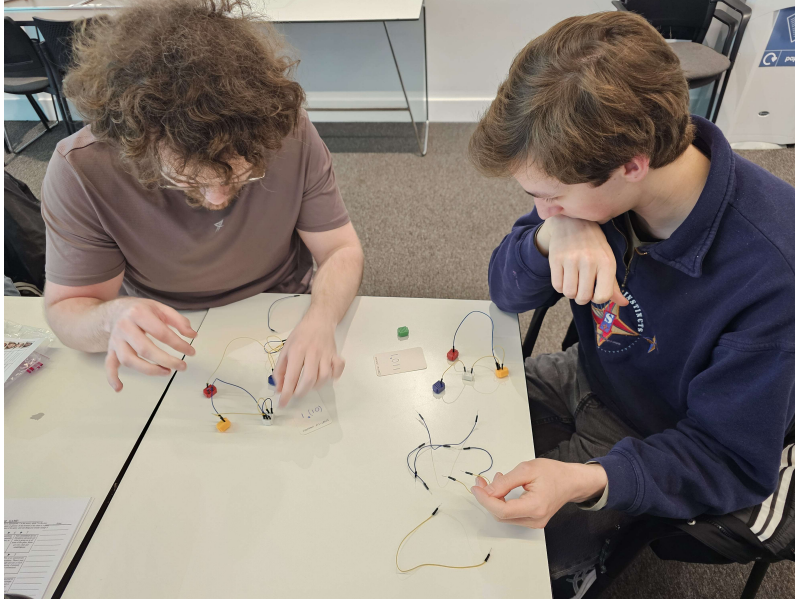


Figure 4: Playtesting of prototype 2.

a DFA from scratch to recognise a particular language exactly is a fun minigame that forms one of the learning objectives. The fastest player also gets to go first, giving a small reward. Discussion is encouraged throughout the rulebook, as Social Learning Theory [11] suggests this facilitates learning.

4 Prototype 2

Prototype 2 was playtested during an in-class activity, as shown in figure 4.

4.1 Go-Again Mechanic

This mechanic was removed for reasons discussed previously.

4.2 Languages

As the tiles were impractical, the languages were simply written onto cards. While this does mean that only an arbitrary subset of words can appear, the cards are designed to contain a variety of words. In prototype 1, some languages often remained for some time and got stale. Adding a decay mechanic would alleviate this, but with only 3 languages they would often be completely different by a player's next turn. Therefore, the number of languages will be increased to 5. This also increases the skill required as there are more languages to consider.

This decay mechanic is inspired from another 2-player game, Fungi [4]. In this mechanic, the languages are arranged in a queue, and the one furthest from the draw pile decays each turn, with

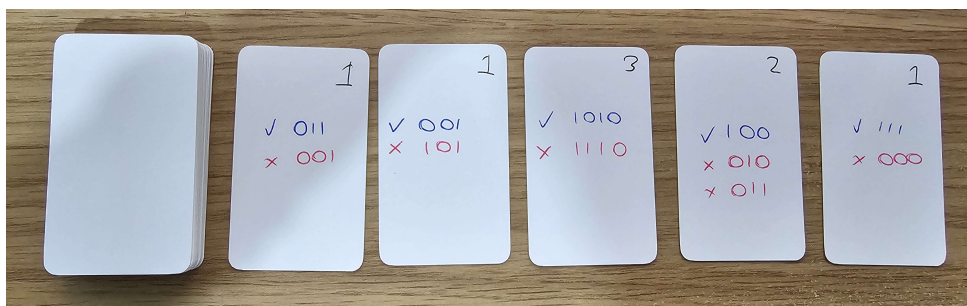


Figure 5: The language queue mechanic with decay.

empty spots being filled so there are always 5 languages. This allows players to see what is upcoming and make more informed decisions, as there are only 2 players so the queue will be similar on their next turn. Experienced players may even consider the opponent's DFA to predict the state of the queue on their next turn. This mechanic is pictured in figure 5.

4.3 Time Limits

A one-minute timer was introduced to keep the game engaging and force the player to think quicker, strengthening their understanding.

4.4 Scoring & Game End

The scoring is based on how many languages are recognised, and how complex they are. This is significantly easier to keep track of with the cards, as players can collect cards that they have recognised, and the game ends when they run out. It is easy to understand and highly linked to the learning outcomes, which is important for constructive alignment [5]. Players who are better at thinking of DFAs in less time will score more points. However, there are some skills beyond the learning objectives for high-level play. Experienced players will think ahead to future moves to leave their DFA in a state that is amenable to recognising future languages. While this doesn't necessarily align with the learning objectives, it demonstrates an exceptional understanding of DFAs and so should be rewarded.

Games that end after a predictable length of time are more practical, especially in a classroom setting. The game can be ended early without giving either player a significant advantage.

5 Final Game

The submitted game is the third prototype, which is not yet physically realised.

5.1 DFA Manipulation

The main issue found was that one move was insufficient to make a meaningful change to the DFA, leading to many stagnant turns. The design ambition is for players to recognise at least one language on most of their turns, so this has been increased to 3 moves per turn. Players also

felt restricted by being unable to change their single accepting state, so an option was introduced for players to swap one state between accepting/rejecting instead of their three moves. Changes to transitions over accepting/rejecting states are incentivised because this leads to more changes which helps players learn about a variety of different DFAs.

5.2 States

Both prototypes suffered from practical issues of physically realising the states. In particular, players often forgot which transitions were 1 and 0, which were the inputs and outputs and which states were which. The final game will include 3D printed states with all of the properties appropriately labelled to make the gameplay significantly easier.

5.3 Special Protocols

A replacement for the go-again mechanic is special protocols, a name for powerups. Recognising several languages in the same turn should feel rewarding and exciting, even more so than just gaining more points, therefore protocols are introduced. Protocols have unique abilities which can be played any time or kept for additional points at the end of the game. This means if a player does not enjoy a particular protocol, or pick it up too near the end for it to be useful, they can simply retain it for extra point(s) (they are designed to be more useful when played). This does introduce some randomness as some are deliberately better than others, however players are able to strategise knowing what their cards are, and it has a small impact on the game overall. It also enhances the potential for a losing player to still win.

6 Card Design

The cards (figure 6) have a blue background that matches with the box art (figure 7), both designed to look like a blueprint, creating a feeling of a creative, methodical game. The points are clearly emphasised within a hexagon using vibrant colours that differs for the number of points, to draw attention to high-scoring cards. Accepted or rejected words are done with both colour and icons, as this aids quicker interpretation of the information [8] and reduces cognitive load [10]. The card is reversible so that both players can sit opposite and have the queue in the middle of the table. It also makes shuffling the cards easier.

7 Conclusion and Future Work

Overall, *Automata Architects* is a successful board game to teach DFAs to any audience, but is particularly applicable at University-level. Modifying a DFA to recognise languages in a queue provides an interesting dynamic whereby players have to plan ahead and strategise to recognise the most languages and win. The scoring system is inherently tied to the learning objectives, and the game will not become stale due to the wealth of different possibilities and special protocol cards.

For future work, the states need to be realised and then some additional playtesting is required. In particular, a study should be conducted on students who are learning about DFAs to test performance in an exam before/after the game is used. The game can also be adapted into a video game, which would automatically determine which languages are recognised.



(a) 1 point. (b) 2 points. (c) 3 points.

Figure 6: Example language cards. Backgrounds are generated by ChatGPT [2].



Figure 7: Box art. Generated by ChatGPT [2].

AI Usage

With repeated interrogation, ChatGPT [2] was used to generate:

- The name of the game.
- The box art.
- The sell sheet background image.
- The background image used in the cards and the rulebook.

References

- [1] John Carroll and Darrell Long. *Theory of finite automata: with an introduction to formal languages*. Prentice-Hall Inc., 1989.
- [2] *ChatGPT*. URL: <https://chatgpt.com/>.
- [3] *CS143: Logic and Automata*. URL: <https://warwick.ac.uk/fac/sci/dcs/teaching/modules/cs143>.
- [4] *Fungi*. URL: <https://www.shutupandsitdown.com/games/fungi/>.
- [5] Catherine Tang John Biggs. *Teaching for Quality Learning at University (4th ed.)* Open University Press, 2011.
- [6] D. R. Krathwohl. “A revision of bloom’s taxonomy: An overview”. In: *In Theory into Practice* (2002).
- [7] Sanjiang Li and Xueqing Yan. “Let’s Play Mahjong!” In: *CoRR* abs/1903.03294 (2019). arXiv: 1903.03294. URL: <http://arxiv.org/abs/1903.03294>.
- [8] Allan Paivio and Kalman Csapo. “Picture superiority in free recall: Imagery or dual coding?” In: *Cognitive Psychology* 5.2 (1973), pp. 176–206. ISSN: 0010-0285. DOI: [https://doi.org/10.1016/0010-0285\(73\)90032-7](https://doi.org/10.1016/0010-0285(73)90032-7). URL: <https://www.sciencedirect.com/science/article/pii/0010028573900327>.
- [9] Yasser Seirawan. *Winning Chess Openings*. Everyman Chess, 2003.
- [10] John Sweller. “Cognitive load during problem solving: Effects on learning”. In: *Cognitive Science* 12.2 (1988), pp. 257–285. ISSN: 0364-0213. DOI: [https://doi.org/10.1016/0364-0213\(88\)90023-7](https://doi.org/10.1016/0364-0213(88)90023-7). URL: <https://www.sciencedirect.com/science/article/pii/0364021388900237>.
- [11] L.S. Vygotsky. *Mind and Society: The Development of Higher Psychological Processes*. Harvard University Press, 1978.

A Prototype 2 - Rulebook

The rulebook for prototype 2 is in figure 8.

DFA Game (v1)

Players 2
Game length Customisable / TBD

What's in the box?

- Language cards.
- Starting language cards.
- 2x5 states of different colours.
- 20 wires.

Setup

1. Divide the states in half, so that each player gets one of each colour.
2. Give each player 5 yellow and 5 blue wires.
3. Choose a colour to represent the start state and a colour to represent the accepting state, depending on what you both find easiest to distinguish.
4. Do the starting minigame (below). The player who wins goes first.
5. Shuffle the language cards and deal out 5 face-up, in a line next to the pile.

Starting Minigame

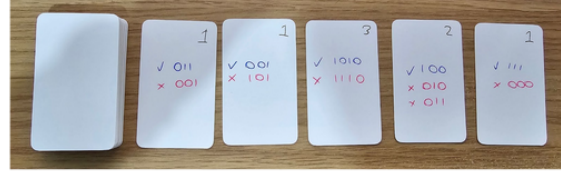
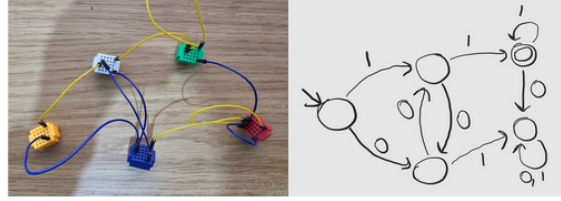
Deal out one 'starting language' card to each player. Both players turn these over simultaneously, and have to build a DFA that accepts exactly this language as quickly as possible, using all of their states. Once the first player is done, they declare this and step back. The other player uses as much time as they need to finish their DFA.

Once both players are done, check if they are correct. The first correct player wins. If both are incorrect, do the minigame again. If one of them is incorrect, fix it. You start playing from the DFA you end up with from this minigame.

Building a DFA & Recognising Languages

A DFA is constructed by connecting wires between the states. A yellow wire represents a 1 and a blue wire represents a 0. I recommend putting the outputs on the outer 2 edges of the DFA and filling up the inputs from the middle to distinguish them. Please note that every state needs to have exactly one yellow and one blue wire coming out of it for it to be a DFA.

A language is recognised if all of the words with a tick are accepted by your DFA and all of the words with a cross are rejected. Languages with a regex must be recognised exactly, for arbitrary length strings.



This example DFA recognises the second and fifth language cards, so this player would pick up both cards and score 2 points total.

On Your Turn

Each turn, you can make one change to your DFA. This constitutes moving one of the outputs from one state to a different state. You can also choose to pass and not make any changes. At the end of your turn, determine which language(s) you have recognised and pick them up. Move the rightmost language card (furthest from the draw pile) into the discard pile. Shuffle the cards along and turn more over so that 5 cards are now visible for the next player's turn.

Scoring & Game End

The game ends when there are no more languages left (or you can end it early if you need to). Count up how many points you have scored from your languages - most points wins.

Timing

After the 1st game, consider using a 1 minute turn timer. Players who run out of time must pass.

(a) Page 1

(b) Page 2

Figure 8: Rulebook for prototype 2

B Cards List

Table 2 shows the types of language cards in the game. Table 3 shows the protocol cards (3 of each).

Points	Description	Count
1	One accept (3 bits), One reject (3 bits)	24
2	Two accept (3 bits), One reject (3 bits)	8
2	One accept (3 bits), Two reject (3 bits)	8
3	One accept (4 bits), One reject (4 bits)	8

Table 2: Types of language cards present in the game.

Points	Title	Ability
1	Time Increase	Double the timer for this turn only.
1	Moves Increase	Make one extra move this turn.
1	Rollover	Stop the timer. Any remaining time goes into a bank that can be used at any time.
2	Destruction	Choose 1-5 face-up cards, remove them from play, and deal back up to 5 cards.
2	Rewind	Stop the timer, then discard all 5 cards and deal out 5 new ones. Start the timer again from the beginning.
2	Cards Increase	Turn over another 3 cards from the top of the deck. At the end, keep any you've recognised and discard the rest.
3	Overhaul	You can make as many changes as you want this turn.

Table 3: Protocol cards in the game (3 copies of each).